

Pembangunan Sistem Freelance Marketplace Untuk Bidang Pengembangan Perangkat Lunak Berbasis Web

Arka Fadila Yasa¹, Denny Sagita Rusdianto², Komang Candra Brata³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹arkayasa@student.ub.ac.id, ²denny.sagita@ub.ac.id, ³k.candra.brata@ub.ac.id

Abstrak

Pertumbuhan sektor teknologi informasi di Indonesia tergolong cepat dan akan terus berkembang, hal tersebut mempengaruhi cara suatu badan usaha untuk memenuhi kebutuhan sumber daya manusia untuk menyelesaikan suatu proyek yang dijalankan. Pekerja lepas atau freelancer merupakan salah satu solusi bagi pemilik proyek yang menginginkan karyawan untuk memenuhi kebutuhan sumber daya manusia. Salah satu faktor penting untuk menentukan kesuksesan proyek yang dikerjakan oleh freelancer adalah merekrut orang yang tepat untuk mengerjakan proyek, namun terdapat permasalahan karena klien pemilik pekerjaan perlu membandingkan profil freelancer satu dengan yang lain. Sehingga untuk mempermudah proses perekrutan, penulis berinisiatif untuk membuat sistem untuk mempertemukan antara pihak yang sedang membutuhkan pekerja lepas untuk mengerjakan suatu proyek dengan freelancer yang memiliki keahlian yang berfokus pada pencarian pekerja di dalam bidang pengembangan perangkat lunak. Pembatasan bidang pelayanan sistem di bidang pengembangan perangkat lunak dikarenakan kebutuhan *freelancer* pada bidang tersebut merupakan yang paling besar. Berdasarkan hasil analisis kebutuhan yang bersumber dari hasil proses elisitasi kebutuhan yang menggunakan metode observasi terhadap sistem serupa yang telah ada, didapatkan lima puluh kebutuhan fungsional sistem, kemudian pada hasil perancangan sistem terdapat dua puluh satu kelas sistem untuk di implementasikan. Hasil pengujian sistem yang telah selesai di implementasikan yang terdiri dari pengujian unit, pengujian integrasi, pengujian validasi, dan pengujian kompatibilitas menunjukkan bahwa sistem sudah dapat berjalan sesuai dengan kebutuhan serta mampu berjalan pada berbagai peramban web.

Kata kunci: *freelancer*, proyek, pengembangan, perangkat lunak, *freelance marketplace*, web.

Abstract

The growth of the information technology sector in Indonesia is relatively fast and will continue to grow, it affects the way a company to meet the needs of human resources to complete a project. Freelancers is one solution for project owners who want employees to meet the needs of human resources. One important factor to determine the success of projects done by freelancers is to recruit the right people to work on the project, but there is a problem because clients need to compare the profiles of freelancers to one another. In the purpose to simplify the recruitment process, the authors took the initiative to create a system to bring together those who are in need of freelancers to work on a project with freelancers who have expertise with focus on finding workers in the field of software development. System are restricted to serve in the field of software development is because the need for freelancers in that field is the greatest one. Based on the results of the needs analysis sourced from the results of the elicitation of needs using the observation method of a similar system that already exists, obtained fifty functional system requirements, then the results of system design there are twenty-one system classes to be implemented. The results of the system testing that have been completed are implemented consisting of unit testing, integration testing, validation testing, and compatibility testing showing that the system can run as needed and is capable of running on various web browsers.

Keywords: *freelancers*, project, development, software, *freelance marketplace*, web.

1. PENDAHULUAN

Pekerja lepas atau istilah populernya disebut *freelancer* merupakan salah satu solusi bagi perusahaan yang menginginkan pegawai untuk memenuhi kebutuhan sumber daya manusia dalam jangka waktu tertentu. Pilihan untuk menjadi *freelancer* saat ini sedang menunjukkan tren yang signifikan. Sedangkan di sisi lain, penggunaan pekerja lepas dapat memberikan manfaat lebih bagi perusahaan yang membutuhkan sumber daya manusia dalam hal memaksimalkan produktivitas karyawan yang sejalan dengan efisiensi biaya untuk merekrut, berbagi wawasan antar pekerja, dan fleksibilitas waktu dalam melakukan pekerjaan. Pemanfaatan pekerja lepas juga dapat dimaksimalkan oleh perusahaan berkat kemajuan teknologi informasi agar dapat memenuhi kebutuhan sumber daya manusia terutama pada keahlian yang dibutuhkan yang tidak dapat dipenuhi oleh pegawai perusahaan yang sudah tersedia dan bekerja secara *full time* sehingga berdampak pada kepada dunia bisnis karena kemudahan untuk memenuhi kebutuhan sumber daya manusia dengan keterampilan atau keahlian khusus (Mustofa, 2018).

Salah satu faktor penting untuk menentukan kesuksesan proyek yang dikerjakan oleh *freelancer* adalah merekrut orang yang tepat untuk mengerjakan proyek, namun permasalahan yang ditemukan pada saat menyeleksi pekerja yang cocok adalah sulitnya pihak perekrut dalam menilai dan memilih calon pekerja yang berkompeten atau tidak, hal tersebut terjadi karena proses yang cukup memakan waktu ketika pihak perekrut melihat dan membandingkan informasi profil *freelancer* yang bertujuan untuk mencari *freelancer* yang lebih cocok untuk mengerjakan suatu proyek.

Sehingga untuk mempermudah proses perekrutan, penulis berinisiatif untuk membuat sistem *freelance marketplace* yaitu sistem yang mampu mempertemukan antara pihak yang membutuhkan *freelancer* dengan *freelancer* yang memiliki keahlian tertentu untuk menyelesaikan suatu pekerjaan yang mampu membantu pihak perekrut dalam menemukan *freelancer* yang cocok. Sistem *freelance marketplace* disajikan dalam bentuk aplikasi berbasis web serta ruang lingkup pelayanan dibatasi hanya untuk proyek yang berada dalam bidang pekerjaan yang berhubungan dengan bidang pengembangan perangkat lunak.

Berdasarkan penelitian yang dilakukan oleh Yuan Tian, Dinusha Wijedasa, David Lo, dan Claire Le Goues untuk membuat sebuah sistem rekomendasi untuk menentukan pemrogram yang mampu memperbaiki suatu kerusakan perangkat lunak yang terdapat pada laporan kerusakan dengan menggabungkan pendekatan *activity based recommendation* dengan *location based recommendation* (Tian, Wijedasa, Lo, & Le Goues, 2016). Penulis mengimplementasikan pendekatan *activity based recommendation* dikarenakan *location based* tidak memungkinkan untuk diterapkan dikarenakan informasi pekerjaan baru dan riwayat pekerjaan yang dilakukan merupakan informasi tekstual, pada *activity based recommendation* digunakan *algoritma cosine similarity* untuk menentukan pekerja yang tepat untuk mengerjakan pekerjaan sebagaimana pada penelitian tersebut menggunakan algoritma yang sama untuk membuat rekomendasi urutan pemrogram yang cocok untuk menangani laporan kerusakan berdasarkan riwayat laporan kerusakan yang telah ditangani.

2. LANDASAN KEPUSTAKAAN

2.1 Freelancer

Pekerja lepas atau biasa disebut dengan istilah inggris populernya *freelancer* adalah pekerja yang mengabdikan kepada klien pada jangka waktu tertentu sesuai dengan ketentuan pada kontrak. Di Indonesia, bekerja dengan menjadi *freelancer* bukan lagi merupakan hal baru. Sudah banyak orang yang bekerja dengan cara ini dengan menjadi penerjemah, konsultan, penulis, dan lain-lain. Namun, pada saat itu menjadi *freelancer* kurang begitu dikenal hal itu ditambah dengan tersedianya fasilitas jaminan sosial seperti yang dimiliki karyawan perusahaan pada umumnya. Dengan dukungan perkembangan teknologi pada bidang informasi dan komunikasi, *freelancer* menjadi umum digunakan sehingga lebih banyak badan usaha yang memanfaatkan pekerja lepas terutama pada perusahaan mikro dan keluarga (Anggrian & Sumarlin, 2016).

2.2 Freelance Marketplace

Pengertian dari *freelance marketplace* adalah suatu *platform* daring yang mampu mengkoneksikan antara para pekerja lepas (*freelancer*) dengan pemilik bisnis yang

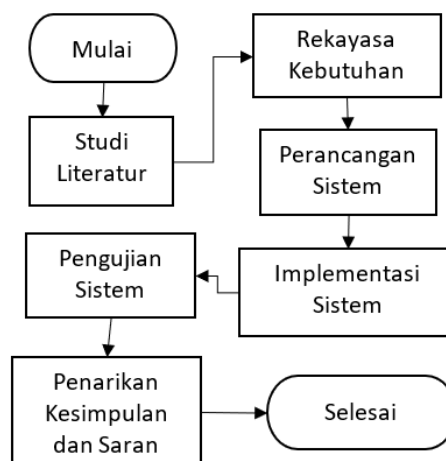
menggunakan jasanya baik dalam bentuk individual hingga perusahaan besar. Dimana *platform* ini ditunjang oleh situs web yang mampu memberikan informasi detail mengenai *freelancer* termasuk riwayat pekerjaan dan kualifikasi yang dimiliki. Cara kerja dari *platform* ini pada dasarnya adalah pemilik proyek mempublikasikan tawaran pengerjaan proyek kemudian *freelancer* yang tertarik untuk mengerjakannya membalas tawaran tersebut. Dalam layanan ini harus terdapat kepastian bahwa pengguna jasa harus mendapatkan hasil pengerjaan yang dia bayarkan untuk itu sedangkan *freelancer* harus mendapatkan pembayaran ketika selesai menyelesaikan pekerjaannya (Bacon, Chen, Parkes, & Rao, 2009).

2.3 Cosine Similarity

Cosine Similarity adalah algoritma yang mampu menghitung tingkat kecocokan antara dua dokumen yang dibandingkan. *Cosine Similarity* menggunakan *Term Frequency (TF)* yang akan menjadikan suatu dokumen menjadi kumpulan (*term*) beserta frekuensi kemunculan, sehingga dapat ditemukan kemiripannya (Singhal, 2001). Sehingga fungsi *similarity* mampu untuk mencari kemiripan antara dua dokumen yang dibandingkan dalam skala 0 hingga 1. Penentuan tingkat kemiripan suatu kadokumen dengan kumpulan kata yang ingin dicari kemiripannya yang disebut dengan *query* dapat menggunakan angka *similarity measure*. Semakin besar nilai angka *similarity measure* maka semakin mirip dokumen dengan *query* yang dibandingkan.

3. METODOLOGI PENELITIAN

Pada kegiatan penelitian yang dilakukan terdapat metodologi yang meliputi beberapa tahapan yang berjalan secara berurutan mulai dari tahap studi literatur untuk mengumpulkan dasar teori yang diperlukan untuk pelaksanaan penelitian, rekayasa kebutuhan untuk mengumpulkan kebutuhan sistem, perancangan sistem untuk membuat rancangan sebagai dasar implementasi, kemudian implementasi sistem untuk menerapkan rancangan hingga menjadi sistem utuh, pengujian sistem untuk memastikan bahwa sistem dapat berjalan baik dan sesuai, hingga penarikan kesimpulan dan saran. Diagram metodologi penelitian yang dilakukan dilihat pada gambar 1.

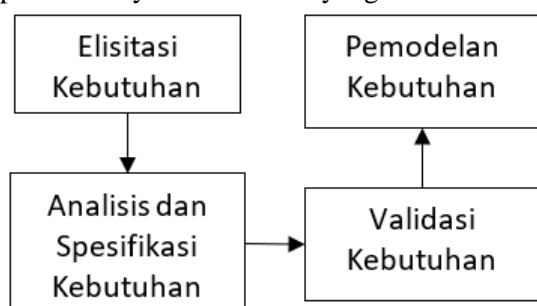


Gambar 1. Diagram metodologi penelitian

Tahapan studi literatur dilakukan untuk mencari dan mempelajari dasar teori yang berkaitan dalam rangka menyelesaikan masalah yang menjadi dasar dari penelitian dilakukan. Pada tahap rekayasa kebutuhan, kegiatan elisitasi kebutuhan dilaksanakan dengan metode observasi pada sistem *freelance marketplace* yang sudah tersedia yaitu situs *sribulancer.com*, hasil dari proses ini disajikan dalam bentuk daftar fitur sistem yang akan dikembangkan. Dari hasil proses elisitasi kebutuhan kemudian dilakukan analisis kebutuhan menggunakan metode *Object Oriented Analysis* yang kemudian dapat dimodelkan proses bisnis yang terjadi pada sistem dalam bentuk *Business Process Model (BPM)* dan daftar kebutuhan perangkat lunak. Kemudian masuk ke tahap perancangan sistem yang bertujuan untuk menghasilkan rancangan sistem yang akan dibuat agar sesuai dengan kebutuhan yang sudah didefinisikan sebagai dasar untuk melakukan implementasi. Metode *Object Oriented Design (OOD)* digunakan pada perancangan. Kemudian dilakukan tahap implementasi sistem yang merupakan tahapan untuk menerapkan hasil perancangan yang dibuat hingga menjadi sistem yang utuh dengan menerapkan metode *Object Oriented Programming (OOP)*. Kemudian masuk ke tahap pengujian untuk mengetahui kesesuaian sistem dan kualitas. Pengujian dilakukan mulai dari tingkat unit, integrasi, hingga validitas fungsional serta dilakukan pula pengujian kompatibilitas untuk pengujian non-fungsional sistem. Setelah tahapan-tahapan penelitian telah selesai dilaksanakan maka dilakukan penarikan kesimpulan sebagai jawaban atas rumusan masalah serta saran penelitian lebih lanjut.

4. REKAYASA KEBUTUHAN

Pada proses rekayasa kebutuhan yang dilakukan, secara umum terdiri dari proses elisitasi kebutuhan, kemudian dari hasil proses elisitasi kebutuhan dijadikan dasar untuk melakukan analisis dan spesifikasi kebutuhan, kemudian dilakukan validasi hasil analisis kebutuhan, setelah itu kebutuhan yang didapatkan kemudian dibuat pemodelan agar mudah dipahami. Gambar 2 menunjukkan proses rekayasa kebutuhan yang dilaksanakan.



Gambar 2. Diagram proses rekayasa kebutuhan.

Pada proses ini, dimulai dari proses elisitasi kebutuhan yaitu proses untuk menemukan kebutuhan sebagai dasar untuk mengembangkan sebuah sistem. Proses elisitasi kebutuhan dilakukan menggunakan metode observasi dengan cara melakukan observasi yang dilakukan pada halaman bantuan sistem dan observasi langsung pada tampilan antarmuka web. Dari hasil elisitasi kebutuhan, didapatkan fitur-fitur dasar dari sistem *freelance marketplace* ditunjukkan pada tabel 1.

Tabel 1 Fitur sistem *freelance marketplace*

No.	Fitur
1	Manajemen pekerjaan yang sudah dibuat.
2	Memeriksa pekerjaan baru.
2	Melamar pekerjaan.
3	Rekrutmen pelamar pekerjaan.
4	<i>Workspace</i> pekerjaan.
5	Pembayaran tagihan
6	Manajemen portofolio <i>freelancer</i>
7	Pengaturan informasi profil.
8	Rekomendasi <i>freelancer</i> .

Sehingga berdasarkan dari sistem yang diobservasi maka pada penelitian ini, penulis

berinisiatif untuk membuat sebuah sistem *freelance marketplace* yang akan menyempurnakan sistem yang sudah ada dengan menggunakan fitur rekomendasi kecocokan antara *freelancer* yang melamar proyek dengan deskripsi proyek dengan memanfaatkan algoritma *cosine similarity* dengan memakai metode *activity-based recommendation* yaitu dengan memproses data riwayat proyek yang telah dikerjakan dengan data deskripsi proyek kemudian dilakukan penghitungan nilai *cosine similarity*. Untuk melakukan komunikasi antara *freelancer*, klien, dan administrator digunakan aplikasi perpesanan yang sudah ada yaitu *Whatsapp*. Kebutuhan bidang keahlian yang dapat dipilih dibatasi hanya pada pengembangan web, pengembangan aplikasi mobile, desain aplikasi, dan pengujian aplikasi. Selain itu, dikarenakan sistem *freelance marketplace* digunakan oleh banyak pengguna dengan menggunakan peramban web yang berbeda maka sistem yang akan dibuat harus memiliki kompatibilitas untuk dapat berjalan pada berbagai macam peramban web. Dari hasil fitur-fitur sistem yang didapatkan, maka didapatkan 50 kebutuhan fungsional dan kebutuhan non-fungsional berupa kompatibilitas sistem. Tabel 2 menunjukkan aktor yang terlibat dalam penggunaan sistem.

Tabel 2 Hasil identifikasi aktor

Aktor	Deskripsi
<i>User</i>	Pengguna yang belum melakukan autentikasi pada sistem.
<i>Freelancer</i>	Pengguna yang mengerjakan pekerjaan proyek dari Klien.
Administrator	Pengguna sistem yang melakukan mediasi proses proyek dari awal sampai akhir.
Klien	Pengguna sistem yang membutuhkan pihak <i>freelancer</i> untuk mengerjakan proyek.

Dari hasil analisis kebutuhan dapat dirumuskan poin-poin pernyataan spesifik dari tiap kebutuhan saat spesifikasi kebutuhan. Proses spesifikasi membuat kebutuhan sistem menjadi lebih spesifik dan detail agar lebih jelas dan tidak ambigu serta mudah dipahami yang

membuat kebutuhan lebih konsisten. Pada tiap kebutuhan disematkan kode kebutuhan agar mudah dilakukan pelacakan, kontrol dan identifikasi jika diperlukan. Kemudian kebutuhan sistem dibuat pemodelannya dengan menggunakan *use case diagram* yang terdapat pada gambar 3 sebagai representasi visual dari interaksi antara pengguna dengan sistem yang kemudian dijelaskan mendetail pada *use case scenario*.

5. PERANCANGAN

Perancangan sistem dilakukan berdasarkan kebutuhan sistem yang telah didapatkan pada fase rekayasa kebutuhan. Perancangan sistem meliputi mulai dari perancangan arsitektur yang dilakukan untuk merancang struktur komponen yang terdapat pada sistem dimana terdapat rancangan hubungan pemanggilan antar kelas sistem yang direpresentasikan oleh *sequence diagram* serta hubungan antar kelas sistem yang disajikan dalam bentuk *class diagram* yang membentuk total 21 kelas sistem, kemudian perancangan komponen yang memaparkan secara detail mengenai alur algoritma pada unit atau *method* yang disajikan dalam bentuk *pseudocode*, struktur data yang digunakan pada sistem *Freelance Marketplace* akan dipaparkan secara rinci beserta hubungan antar entitas data direpresentasikan secara visual melalui *Entity Relationship Diagram (ERD)* sehingga berdasarkan hasil perancangan data terdapat sembilan entitas yang terdiri dari enam *strong entity* dan tiga *weak entity*. Perancangan antarmuka merupakan proses merancang tampilan halaman sebagai tempat interaksi pengguna. Gambar 4 menampilkan rancangan antarmuka halaman *viewMerekrutPekerja*.

The screenshot displays a web interface for project management. At the top, 'PROYEK 1' details are shown: Bidang (Magelang), Keahlian (Magelang), Anggaran Proyek (Rp. 10000000), Tanggal Ditutup (dd/mm/yyyy), and Deskripsi. Below this, there are two sections for selecting freelancers. The 'Terpilih' (Selected) section shows a freelancer named 'Nama Lengkap (Username)' with details: Tawaran harga (Rp. 1000000), Domisili (Magelang, Indonesia), Bidang (Magelang), Keahlian (Magelang), and Alasan melamar (Magelang). The 'Belum Terpilih' (Not Selected) section shows a similar form for a freelancer named 'Freelancer ini memiliki kecukupan 100% terhadap proyek' with details: Tawaran harga (Rp. 1000000), Domisili (Magelang, Indonesia), Bidang (Magelang), Keahlian (Magelang), and Alasan melamar (Magelang). Buttons for 'Detail Profil' and 'Rekrut' are visible next to each freelancer entry.

Gambar 4. Hasil perancangan antarmuka *viewMerekrutPekerja*

6. IMPLEMENTASI

Implementasi merupakan kegiatan untuk menerapkan perancangan yang telah dibuat hingga menjadi suatu sistem yang utuh dan dapat beroperasi. Implementasi kode program sistem dilakukan dengan menggunakan bahasa pemrograman *PHP* dengan *framework* pengembangan *CodeIgniter*. Kemudian implementasi antarmuka dengan *HTML*, *CSS*, *Javascript*, dan *Framework Bootstrap*. Implementasi data dengan *MySQL*. Ditunjukkan pada gambar 5 yang menunjukkan tampilan antarmuka *viewUlasan* merupakan salah satu hasil implementasi sistem agar pengguna dapat memberikan ulasan kepada hasil pekerjaan *freelancer* yang bekerja.

The screenshot shows a web form titled 'Berikan Ulasan Freelancer'. It contains a section for 'Freelancer satu' with fields for 'Username : freasone' and 'Berikan nilai bintang terhadap pekerjaan saya.' (Give a star rating to my work). Below this is a star rating system with five stars. There is also a section for 'Tuliskan ulasan' (Write a review) with a text area and a 'Masukkan ulasan anda' (Enter your review) prompt. A green button labeled 'Beri Ulasan' (Give Review) is at the bottom.

Gambar 5. Hasil implementasi antarmuka *viewUlasan*

7. PENGUJIAN

Pengujian bertujuan untuk mencari kerusakan sistem ketika dijalankan dan memastikan sistem sesuai dengan kebutuhan. Pelaksanaan pengujian dilakukan oleh penulis menggunakan metode *manual testing* yaitu dengan berinteraksi langsung dengan sistem tanpa menggunakan bantuan perangkat lain. Pengujian unit ditujukan pada unit terkecil pada sistem yaitu *class* dimana objek pengujian adalah *testable unit* dalam bentuk *method* atau fungsi pada *class*, strategi pengujian *white-box* dengan teknik pengujian *basis path testing* digunakan pada pengujian ini. Kemudian pengujian pada jalur-jalur yang ditemukan dilakukan dengan memanggil *method* yang diuji melalui kelas *TestDriver* yang dibuat. Pengujian integrasi dilakukan untuk menguji hubungan antar unit ketika dijalankan yang bertujuan untuk menemukan kesalahan saat interaksi antar unit terjadi. Pengujian dilakukan menggunakan strategi pengujian *white-box* dengan teknik *basis path testing* kemudian pengujian dilakukan dengan memanggil *method* yang diuji melalui kelas *TestDriver* yang dibuat. Pengujian validasi dilakukan untuk



menguji kesesuaian sistem dengan kebutuhan fungsional serta menemukan kecacatan pada sistem. Strategi pengujian menggunakan *black-box* dengan teknik *scenario-based testing* yang berdasarkan pada *use case scenario* sebagai dasar untuk membuat kasus uji. Hasil dari pengujian unit, integrasi, dan validasi menunjukkan bahwa perangkat lunak sudah memenuhi kebutuhan dan dapat berjalan dengan baik. Sedangkan pada kebutuhan non-fungsional dilakukan pengujian kompatibilitas menggunakan perangkat lunak *Sortsite* sebagai kakas bantu pengujian. Hasil pengujian kompatibilitas menunjukkan bahwa sistem yang dibuat mampu berjalan pada berbagai peramban web.

8. KESIMPULAN

Dari kegiatan penelitian yang telah dilakukan dapat ditarik beberapa kesimpulan bahwa dari analisis kebutuhan dengan menggunakan teknik observasi didapatkan 50 kebutuhan fungsional sistem untuk membangun sistem *freelance marketplace* yang mampu menghitung kecocokan antara *freelancer* dengan proyek dengan menggunakan algoritma *cosine similarity*.

Kemudian dari tahap perancangan sistem, terdapat 21 kelas sistem untuk membangun sistem *freelance marketplace* yang mampu menghitung kecocokan antara *freelancer* dengan proyek yang kemudian di implementasikan dalam kode program dan terdapat sembilan entitas yang terlibat yang terdiri dari enam *strong entity* dan tiga *weak entity* pada perancangan data yang kemudian di implementasikan pada *database management system MySQL*.

Setelah implementasi hingga menjadi sistem yang utuh dan dapat berjalan selesai dilakukan, maka dilakukan pengujian fungsional sistem mulai dari unit, integrasi, dan validasi serta ditambah dengan pengujian kompatibilitas. Hasil dari setiap pengujian menunjukkan hasil valid pada setiap kasus uji, sehingga bisa dikatakan sistem sudah memenuhi semua kebutuhan yang telah ditentukan. Serta hasil pengujian kompatibilitas yang telah dilakukan, dapat disimpulkan bahwa sistem sudah dapat dijalankan pada berbagai peramban.

Saran untuk pengembangan penelitian lebih lanjut dapat ditambahkan fitur untuk mempermudah komunikasi antara klien dengan

freelancer serta moderasi dari administrator beberapa diantaranya adalah penambahan fitur *live chat* di dalam sistem dan integrasi dengan layanan penyimpanan yang telah ada.

9. DAFTAR PUSTAKA

- Anggrian, W. M., & Sumarlin, A. W. (2016). Pengaruh Komitmen Tenaga Kerja Lepas terhadap Motivasinya dalam Perusahaan Keluarga di Sektor Informal. *Jurnal Manajemen Teknologi*, 15(2), 139–153. <https://doi.org/10.12695/jmt.2016.15.2.3>
- Bacon, D. F., Chen, Y., Parkes, D., & Rao, M. (2009). A market-based approach to software evolution. In *Proceedings of the Conference on Object-Oriented Programming Systems, Languages, and Applications, OOPSLA* (hal. 973–980). <https://doi.org/10.1145/1639950.1640066>
- Mustofa. (2018). Pekerja Lepas (Freelancer) Dalam Dunia Bisnis. *Jurnal Mozaik*, 10(1), 19–25. <https://doi.org/10.1017/CBO9781107415324.004>
- Singhal, A. (2001). Modern Information Retrieval: A Brief Overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 24(4).
- Tian, Y., Wijedasa, D., Lo, D., & Le Goues, C. (2016). Learning to rank for bug report assignee recommendation. *IEEE International Conference on Program Comprehension, 2016-July*, 1–10. <https://doi.org/10.1109/ICPC.2016.7503715>